Network Flows

1. Introduction

1.2 Notation and Definitions
Notation and Definitions

- In this section we give several basic definitions from graph theory and present some basic notation.
- We also state some elementary properties of graphs.
Notation and Definitions

- **Directed graph**
  - A directed graph $G = (N, A)$ consists of a set $N$ of nodes and a set $A$ of arcs whose elements are ordered pairs of distinct nodes.

$N = \{1, 2, 3, 4, 5, 6, 7\}$

$A = \{(1, 2), (1, 3), (2, 3), (2, 4), (3, 6), (4, 5), (4, 7), (5, 2), (5, 3), (5, 7), (6, 7)\}$. 
Notation and Definitions

- **Directed network**
  - A *directed network* is a *directed graph* whose nodes and/or arcs have associated:
    - costs
    - capacities
    - and/or supplies and demands

- We use the terms “graph” and “network” synonymously.

- We let $n$ denote the number of nodes and $m$ denote the number of arcs in $G$. 
Notation and Definitions

**Undirected Graphs**

- In an *undirected graph* arcs are unordered pairs of distinct nodes.

![Diagram of an undirected graph with nodes 1, 2, 3, 4, 5, and 6 connected in a star formation.]

- In an undirected graph, we can refer to an arc joining the node pair $i$ and $j$ as either $(i, j)$ or $(j, i)$. 
Notation and Definitions

- **Tails and Heads**
  - A directed arc \((i, j)\) has two *endpoints* \(i\) and \(j\).
  - Node \(i\) is the *tail* of arc \((i, j)\) and node \(j\) is its *head*.
  - The arc \((i, j)\) *emanates* from node \(i\) and *terminates* at node \(j\).
  - An arc \((i, j)\) is *incident to* nodes \(i\) and \(j\).
  - The arc \((i, j)\) is an *outgoing arc* of node \(i\) and an *incoming arc* of node \(j\).
  - Whenever an arc \((i, j) \in A\), we say that node \(j\) is *adjacent* to node \(i\).
Notation and Definitions

- **Degrees**
  - The *indegree* of a node is the number of incoming arcs of that node and its *outdegree* is the number of its outgoing arcs.
  - The *degree* of a node is the sum of its indegree and outdegree.
  - The sum of indegrees of all nodes equals the sum of outdegrees of all nodes and both are equal to the number of arcs $m$ in the network.
Notation and Definitions

- Example of degree:

  - Node 3 has an indegree of 3, an outdegree of 1, and a degree of 4.
Notation and Definitions

- **Adjacency List**
  - The *arc adjacency list* $A(i)$ of a node $i$ is the set of arcs emanating from that node, $A(i) = \{(i, j) \in A : j \in N\}$.
  - The *node adjacency list* $A(i)$ is the set of nodes adjacent to that node; in this case, $A(i) = \{j \in N : (i, j) \in A\}$.
  - We assume that arcs in the adjacency list $A(i)$ are arranged so that the head nodes of arcs are in increasing order.
Notation and Definitions

- **Multiarcs and Loops**
  - *Multiarcs* are two or more arcs with the same tail and head nodes.
  - A *loop* is an arc whose tail node is the same as its head node.
Notation and Definitions

- **Subgraph**
  - A graph $G' = (N', A')$ is a *subgraph* of $G = (N, A)$ if $N' \subseteq N$ and $A' \subseteq A$.
  - We say that $G' = (N', A')$ is the subgraph of $G$ *induced* by $N'$ if $A'$ contains each arc of $A$ with both endpoints in $N'$.
  - A graph $G' = (N', A')$ is a *spanning subgraph* of $G = (N, A)$ if $N' = N$ and $A' \subseteq A$. 
Notation and Definitions

- **Walk**
  - A walk in a directed graph $G = (N, A)$ is a subgraph of $G$ consisting of a sequence of nodes and arcs
    \[ i_1 - a_1 - i_2 - a_2 - \ldots - i_{r-1} - a_{r-1} - i_r \]
  - satisfying the property that for all $1 \leq k \leq r - 1$,
  - either $a_k = (i_k, i_{k+1}) \in A$ or $a_k = (i_{k+1}, i_k) \in A$.
  - Alternatively, we shall sometimes refer to a walk as a set of (sequence of) arcs (or of nodes) without any explicit mention of the nodes (without explicit explicit mention of arcs).
Notation and Definitions

- (a) 1-2-5-7
- (b) 1-2-4-5-2-3
**Notation and Definitions**

- **Directed Walk**
  - A *directed walk* is an oriented version of a walk in the sense that for any two consecutive nodes $i_k$ and $i_{k+1}$ on the walk, $(i_k, i_{k+1}) \in A$.

- (a) is not a directed walk
- (b) is a directed walk
Notation and Definitions

- **Path**
  - A *path* is a walk without any repetition of nodes.
  - Directions are ignored.

- (a) is a path, 1-2-5-7
- (b) is not a path because it repeats node 2 twice., 1-2-4-5-2-3
Notation and Definitions

- We can partition the arcs of a path into two groups: *forward arcs* and *backward arcs*.

\[
\text{\begin{tikzpicture}
    \node (n1) at (0,0) {1};
    \node (n2) at (1,1) {2};
    \node (n5) at (2,0) {5};
    \node (n7) at (2.5,0) {7};
    \draw[-latex] (n1) -- (n2);
    \draw[-latex] (n2) -- (n1);
    \draw[-latex] (n2) -- (n5);
    \draw[-latex] (n5) -- (n7);
\end{tikzpicture}}
\]

- The arcs (1, 2) and (5, 7) are forward arcs and the arc (5, 2) is a backward arc.
Notation and Definitions

- **Directed Path**
  - A directed path is a directed walk without any repetition of nodes.
  - In other words, a directed path has no backward arcs.
  - Directions are important.
Notation and Definitions

- **Cycle**
  - A cycle is a path \(i_1 - i_2 - \ldots - i_r\) together with the arc \((i_r, i_1)\) or \((i_1, i_r)\).
  - We refer to a cycle using the notation \(i_1 - i_2 - \ldots - i_r - i_1\).
  - Just as we did for paths, we can define forward and backward arcs in a cycle.
  - Directions are ignored.

- The arcs \((5, 3)\) and \((3, 2)\) are forward arcs and the arc \((5, 2)\) is a backward arc of the cycle 2-5-3.
Notation and Definitions

- **Directed Cycle**
  - A directed cycle is a directed path $i_1 - i_2 - \ldots - i_r$ together with the arc $(i_r, i_1)$.
  - Directions are important.

  - (a) is a cycle, but not a directed cycle;
  - (b) is a directed cycle.
Notation and Definitions

- **Acyclic Graph**
  - A graph is a *acyclic* if it contains no directed cycle.
Notation and Definitions

- **Connectivity**
  - We will say that two nodes $i$ and $j$ are *connected* if the graph contains at least *one path* from node $i$ to node $j$.
  - A graph is connected if every pair of its nodes is connected; otherwise, the graph is *disconnected*.

- (a) is connected graph, and (b) is disconnected graph
Notation and Definitions

- **Components**
  - We refer to the maximal connected subgraphs of a disconnected network as its *components*.
  - This graph has two components consisting of the node sets \( \{1, 2, 3, 4\} \) and \( \{5, 6\} \).
Notation and Definitions

- **Strong Connectivity**
  - A connected graph is *strongly connected* if it contains at least one directed path from every node to every other node.

- A strongly connected graph

![Diagram of a strongly connected graph]
Notation and Definitions

- **Cut**
  - A cut is a partition of the node set $N$ into two parts, $S$ and $\overline{S} = N - S$.
  - Each cut defines a set of arcs consisting of those arcs that have one endpoint in $S$ and another endpoint in $\overline{S}$.
  - We refer to this set of arcs as a cut and represent it by the notation $[S, \overline{S}]$. 
A cut with $S = \{1, 2, 3\}$ and $\bar{S} = \{4, 5, 6, 7\}$.

The set of arcs in this cut are $\{(2, 4), (5, 2), (5, 3), (3, 6)\}$. 
**Notation and Definitions**

- **s-t Cut**
  - An *s-t* cut is defined with respect to two distinguished nodes *s* and *t*, and is a cut \([S, \bar{S}]\) satisfying the property that \(s \in S\) and \(t \in \bar{S}\).

- If \(s = 1\) and \(t = 6\), this cut is an *s-t* cut;
- but if \(s = 1\) and \(t = 3\), this cut is not an *s-t* cut.
Notation and Definitions

- **Tree**
  - A *tree* is a connected graph that contains no cycle.

- **Property**
  - (a) A tree on $n$ nodes contains exactly $n - 1$ arcs.
  - (b) A tree has at least two leaf nodes (i.e., nodes with degree 1).
  - (c) Every two nodes of a tree are connected by a unique path.
Notation and Definitions

- Two examples of trees
Notation and Definitions

- **Forest**
  - A graph that contains no cycle is a *forest*.
  - Alternatively, a forest is a collection of trees.

  ![Diagram of a forest](image)

  - an example of a forest.
Notation and Definitions

- **Subtree**
  - A connected subgraph of a tree is a *subtree*. 
Notation and Definitions

- **Rooted Tree**
  - A rooted tree is a tree with a specially designated node, called its *root*; we regard a rooted tree as though it were hanging from its root.

- an instance of a rooted tree; node 1 is the root node.
We often view the arcs in a rooted tree as defining predecessor-successor (or parent-child) relationships.

- Node 5 is the predecessor of nodes 6 and 7,
- Node 1 is the predecessor of nodes 2, 4, and 5.
Notation and Definitions

- Each node $i$ (except the root node) has a unique predecessor, which is the next node on the unique path in the tree from that node to the root;
- We store the predecessor of node $i$ using a predecessor index $\text{pred}(i)$.
- If $j = \text{pred}(i)$, we say that node $j$ is the predecessor of node $i$ and node $i$ is a successor of node $j$.
- These predecessor indices uniquely define a rooted tree and also allow us to trace out the unique path from any node back to the root.
Notation and Definitions

There are two special types of rooted trees, called a *directed in-tree* and a *directed out-tree*. 
Notation and Definitions

- **Directed-Out-Tree**
  - A tree is a *directed out-tree* routed at node $s$ if the unique path in the tree from node $s$ to every other node is a directed path.
  - An instance of a directed out-tree rooted at node 1.
  - Observe that every node in the directed outtree (except node 1) has indegree 1.
Notation and Definitions

- **Directed-In-Tree**
  - A tree is a *directed in-tree* routed at node \( s \) if the unique path in the tree from any node to node \( s \) is a directed path.

- an instance of a directed in-tree rooted at node 1.

- Observe that every node in the directed in-tree (except node 1) has outdegree 1.
Notation and Definitions

- **Spanning Tree**
  - A tree $T$ is a spanning tree of $G$ if $T$ is a spanning subgraph of $G$.
  - Every spanning tree of a connected $n$-node graph $G$ has $(n - 1)$ arcs.
  - We refer to the arcs belonging to a spanning tree $T$ as tree arcs and arcs not belonging to $T$ as nontree arcs.
Notation and Definitions

- two spanning trees of the graph
Notation and Definitions

- **Undirected networks**
  - An undirected arc \((i, j)\) has two endpoints, \(i\) and \(j\), but its tail and head nodes are undefined.
  - If the network contains the arc \((i, j)\), node \(i\) is adjacent to node \(j\), and node \(j\) is adjacent to node \(i\).
The End